

# LEHRVERANSTALTUNGSBESCHREIBUNG

## 1. Angaben zum Programm

|                                 |  |
|---------------------------------|--|
| 1.1 Hochschuleinrichtung        | <b>Babes-Bolyai Universität, Cluj-Napoca</b> |
| 1.2 Fakultät                    | Mathematik und Informatik                    |
| 1.3 Department                  | Informatik                                   |
| 1.4 Fachgebiet                  | Informatik                                   |
| 1.5 Studienform                 | Bachelor                                     |
| 1.6 Studiengang / Qualifikation | Informatik in deutscher Sprache              |

## 2. Angaben zum Studienfach

|   |   |              |   |                   |   |                |             |
|---|---|--------------|---|-------------------|---|----------------|-------------|
| 2.1 LV-Bezeichnung (de)<br>(en)<br>(ro) | Objekt-Orientierte Programmierung<br>Object-oriented Programming<br>Programare orientată obiect |              |   |                   |   |                |             |
| 2.2 Lehrverantwortlicher – Vorlesung    | Dominik Knoll, PhD  |              |   |                   |   |                |             |
| 2.3 Lehrverantwortlicher – Seminar      | Dominik Knoll, PhD  |              |   |                   |   |                |             |
| 2.4 Studienjahr                         | 1   | 2.5 Semester | 2 | 2.6. Prüfungsform | P | 2.7 Art der LV | Pflichtfach |

## 3. Geschätzter Workload in Stunden

|  |     |                             |    |                   |       |
|--|-----|-----------------------------|----|-------------------|-------|
| 3.1 SWS  | 5   | von denen: 3.2<br>Vorlesung | 2  | 3.3 Seminar/Labor | 1+2   |
| 3.4 Gesamte Stundenanzahl im Lehrplan  | 70  | von denen: 3.5<br>Vorlesung | 28 | 3.6 Seminar/Übung | 14+28 |
| Verteilung der Studienzeit:  |     |                             |    |                   | Std.  |
| Studium nach Handbücher, Kursbuch, Bibliographie und Mitschriften                                      |     |                             |    |                   | 20    |
| Zusätzliche Vorbereitung in der Bibliothek, auf elektronischen Fachplattformen und durch Feldforschung |     |                             |    |                   | 20    |
| Vorbereitung von Seminaren/Übungen, Präsentationen, Referate, Portfolios und Essays                    |     |                             |    |                   | 20    |
| Tutorien   |     |                             |    |                   | 14    |
| Prüfungen  |     |                             |    |                   | 6     |
| Andere Tätigkeiten: .....  |     |                             |    |                   | -     |
| 3.7 Gesamtstundenanzahl Selbststudium  | 80  |                             |    |                   |       |
| 3.8 Gesamtstundenanzahl / Semester   | 150 |                             |    |                   |       |
| 3.9 Leistungspunkte  | 6   |                             |    |                   |       |

## 4. Voraussetzungen (falls zutreffend)

|                      |                                 |
|----------------------|---------------------------------|
| 4.1 curricular       | Datenstrukturen und Algorithmen |
| 4.2 kompetenzbezogen | Programmierfähigkeiten          |

## 5. Bedingungen (falls zutreffend)

|   |   |
|---|---|
| 5.1 zur Durchführung der Vorlesung            | Vorlesungsraum, Beamer, Laptop                  |
| 5.2 zur Durchführung des Seminars / der Übung | Labor mit Computern ausgestattet für C++ and QT |

## 6. Spezifische erworbene Kompetenzen

|                                     |   |
|-------------------------------------|---|
| <b>Berufliche<br/>Kompetenzen</b>   | <p>BK1 Kenntnisse von Programmier-Paradigmen und sprach-spezifischen Mechanismen, als auch Identifikation von syntaktischen und semantischen Unterschieden.</p> <p>BK2 Erklärung von bestehenden Software Anwendungen, auf verschiedenen Abstraktionsebenen (Architektur, Klassen, Methoden) unter Verwendung geeigneter Grundkenntnisse.</p> <p>BK3 Erstellen von Quellcode und Testen von Komponenten in einer gegebenen Programmiersprachen, entsprechend vorgegebener Spezifikation.</p> <p>BK4 Testen von Anwendungen basierend auf einem Testplan.</p> <p>BK5 Entwickeln von Programmteilen und begleitender Dokumentation.</p> |
| <b>Transversale<br/>Kompetenzen</b> | <p>TK1 Anwendung der Regeln für gut organisierte und effiziente Arbeit, für verantwortungsvolle Einstellung gegenüber der Lehre und der Wissenschaft, für kreative Förderung des eigenen Potentials, mit Rücksicht auf die Prinzipien und Normen der professionellen Ethik</p> <p>TK3 Anwendung von effizienten Methoden und Techniken für Lernen, Informieren und Recherchieren, für das Entwickeln der Fähigkeiten zur praktischen Umsetzung der Kenntnisse, der Anpassung an die Bedürfnisse einer dynamischen Gesellschaft, der Kommunikation in deutscher Sprache.</p>   |

## 7. Ziele (entsprechend der erworbenen Kompetenzen)

|   |   |
|---|---|
| 7.1 Allgemeine Ziele der Lehrveranstaltung  | Erlernen der objektorientierten Programmierung, sowie der Sprache C++ und der QT Bibliothek   |
| 7.2 Spezifische Ziele der Lehrveranstaltung | <ul style="list-style-type: none"> <li>• Der Unterschied zwischen prozeduralen Programmierung und objektorientierter Programmierung</li> <li>• Verstehen der Klassen als Grundstrukturen der Programmierung</li> <li>• Kenntnis von Software-Architektur</li> </ul> |

## 8. Inhalt

| 8.1 Vorlesung   | Lehr- und Lernmethode                             | Anmerkungen |
|---|---|-------------|
| 1. Überblick <ul style="list-style-type: none"> <li>• Grundlagen von C</li> <li>• Lexikale Elemente</li> <li>• Datentypen, Variablen, Konstanten</li> </ul>                                       | Vortrag, Erklärung, Debatte, praktische Beispiele |             |
| 2. Modulare Programmierung in C++ <ul style="list-style-type: none"> <li>• Funktionen, Parameter</li> <li>• Header Dateien, Bibliotheken</li> <li>• Zeiger, Speicherlayout</li> </ul>             | Vortrag, Erklärung, Debatte, praktische Beispiele |             |
| 3. Die Programmiersprache C++ <ul style="list-style-type: none"> <li>• Standards</li> <li>• C++ Core Guidelines</li> <li>• OOP Prinzipien</li> </ul>  | Vortrag, Erklärung, Debatte, praktische Beispiele |             |
| 4. Objektorientierte Programmierung in C++ <ul style="list-style-type: none"> <li>• Klassen und Objekte</li> <li>• Attribute, Konstruktoren</li> <li>• Vererbung, Substitutionsprinzip</li> </ul> | Vortrag, Erklärung, Debatte, praktische Beispiele |             |
| 5. UML Darstellung, Generische Programmierung, Containerklassen, Iteratoren   | Vortrag, Erklärung, Debatte, praktische Beispiele |             |
| 6. Resource Management (Memory) in C++  | Vortrag, Erklärung, Debatte, praktische Beispiele |             |

|  |   |  |
|--|---|--|
| 7. Polymorphismus, dynamische Bindung, abstrakte Klassen | Vortrag, Erklärung, Debatte, praktische Beispiele |  |
| 8. Typumwandlung, Funktionszeiger, Lambda-Ausdrücke      | Vortrag, Erklärung, Debatte, praktische Beispiele |  |
| 9. Code Qualität, Benutzeroberflächen, QT Überblick      | Vortrag, Erklärung, Debatte, praktische Beispiele |  |
| 10. Core Guidelines, UML Darstellung, Entwurfsmuster     | Vortrag, Erklärung, Debatte, praktische Beispiele |  |
| 11. Entwurfsmuster                                       | Vortrag, Erklärung, Debatte, praktische Beispiele |  |
| 12. Modularisierung                                      | Vortrag, Erklärung, Debatte, praktische Beispiele |  |
| 13. POS Anwendung  | Vortrag, Erklärung, Debatte, praktische Beispiele | Je nach Bedarf, werden die Themen in den letzten zwei Vorlesungen erweitert oder ausgelassen |
| 14. Wiederholung   | Vortrag, Erklärung, Debatte, praktische Beispiele |  |

#### Literatur

1. B. Stroustup, The C++ Programming Language, Addison Wesley, 1998.
2. Bruce Eckel, Thinking in C++, [www.bruceeckel.com](http://www.bruceeckel.com)
3. Alexandrescu, Programarea moderna in C++. Programare generica si modele de proiectare aplicata, Editura Teora, 2002
4. M. Frentiu, B. Parv, Elaborarea programelor. Metode si tehnici moderne, Ed. Promedia, Cluj-Napoca, 1994.
5. E. Horowitz, S. Sahni, D. Mehta, Fundamentals of Data Structures in C++, Computer Science Press, Oxford, 1995.
6. K.A. Lambert, D.W. Nance, T.L. Naps, Introduction to Computer Science with C++, West Publishing Co., New-York, 1996.
7. L. Negrescu, Limbajul C++, Ed. Albastra, Cluj-Napoca 1996.

#### In deutscher Sprache:

1. G. Goos, W. Zimmermann, Objektorientiertes Programmieren und Algorithmen, Springer, Berlin, Heidelberg, New York, 2006.
2. Pötzsch-Heffter, A., Konzepte objektorientierter Programmierung, Springer, Berlin, Heidelberg, 2009.
3. Küchlin, W, Weber, A., Einführung in die Informatik, Objektorientiertes Programmieren mit Java, Springer, Berlin, Heidelberg, New York, 2004.

| 8.2 Seminar   | Lehr- und Lernmethode   | Anmerkungen |
|---|-------------------------|-------------|
| 1. Einfache Aufgaben in C, lokale und globale Variablen, Vektoren und Strukturen. | Beispiele, Diskussionen |             |
| 2. Ein-/Ausgabe, Arrays und Strukturen  | Beispiele, Diskussionen |             |
| 3. Fehlerbehandlung   | Beispiele, Diskussionen |             |
| 4. Klassen, Dynamische Vektoren, Iteration  | Quiz                    |             |
| 5. Abstrakte Klassen, Template Klassen, Polymorphismus                            | Beispiele, Diskussionen |             |
| 6. Architektur  | Beispiele, Diskussionen |             |
| 7. GUI mit Qt   | Quiz                    |             |

|   |                                     |  |
|---|-------------------------------------|--|
| <b>Diagramme</b>  |                                     |  |
| <b>Literatur</b>  |                                     |  |
| 1. B. Stroustup, The C++ Programming Language, Addison Wesley, 1998.<br>2. Bruce Eckel, Thinking in C++, www.bruceeckel.com<br>3. Alexandrescu, Programarea moderna in C++. Programare generica si modele de proiectare aplicate, Editura Teora, 2002<br>4. M. Frentiu, B. Parv, Elaborarea programelor. Metode si tehnici moderne, Ed. Promedia, Cluj-Napoca, 1994.<br>5. E. Horowitz, S. Sahni, D. Mehta, Fundamentals of Data Structures in C++, Computer Science Press, Oxford, 1995.<br>6. K.A. Lambert, D.W. Nance, T.L. Naps, Introduction to Computer Science with C++, West Publishing Co., New-York, 1996.<br>7. L. Negrescu, Limbajul C++, Ed. Albastra, Cluj-Napoca 1996. |                                     |  |
| <b>8.3 Labor</b>  |                                     |  |
| 1. Visual Studio Installation. Spezifikation, Design und Implementierung einfacher Aufgaben in C++.   | Beispiele, Diskussionen             |  |
| 2. Modulare Programmierung in C++   | Beispiele, Diskussionen             |  |
| 3. Einfache Klassen und mathematische Operationen   | Beispiele, Diskussionen             |  |
| 4. Konsolen Ein- und Ausgabe und Tests  | Beispiele, Diskussionen             |  |
| 5. Komplexe Klasse  | Beispiele, Diskussionen             |  |
| 6. Unit Tests   | Beispiele, Diskussionen             |  |
| 7. Anwendung in 3 Tier Architektur  | Beispiele, Diskussionen             |  |
| 8. Modularisierung  | Beispiele, Diskussionen             |  |
| 9. Anwendung in 3 Tier Architektur  | Beispiele, Diskussionen, Teamarbeit |  |
| 10. Unit Tests  | Beispiele, Diskussionen, Teamarbeit |  |
| 11. Datei Ein- und Ausgabe  | Beispiele, Diskussionen, Teamarbeit |  |
| 12. Modularisierung   | Beispiele, Diskussionen, Teamarbeit |  |
| 13. Anwendung mit Qt Oberfläche   | Beispiele, Diskussionen, Teamarbeit |  |
| 14. Abgabe Laborarbeiten  | Beispiele, Diskussionen, Teamarbeit |  |
| <b>Literatur:</b>   |                                     |  |
| 1. B. Stroustup, The C++ Programming Language, Addison Wesley, 1998.<br>2. Bruce Eckel, Thinking in C++, www.bruceeckel.com<br>3. Alexandrescu, Programarea moderna in C++. Programare generica si modele de proiectare aplicate, Editura Teora, 2002<br>4. M. Frentiu, B. Parv, Elaborarea programelor. Metode si tehnici moderne, Ed. Promedia, Cluj-Napoca, 1994.<br>5. E. Horowitz, S. Sahni, D. Mehta, Fundamentals of Data Structures in C++, Computer Science Press, Oxford, 1995.<br>6. K.A. Lambert, D.W. Nance, T.L. Naps, Introduction to Computer Science with C++, West Publishing Co., New-York, 1996.<br>7. L. Negrescu, Limbajul C++, Ed. Albastra, Cluj-Napoca 1996. |                                     |  |

## 9. Verbindung der Inhalte mit den Erwartungen der Wissensgemeinschaft, der Berufsverbände und der für den Fachbereich repräsentativen Arbeitgeber

Diese Vorlesung wird an international bekannten Universitäten im Fachgebiet Informatik angeboten. Der Inhalt der Vorlesung ist wichtig für die Softwarefirmen und entspricht der ACM Richtlinien.

## 10. Prüfungsform

| Veranstaltungsart  | 10.1 Evaluationskriterien   | 10.2 Evaluationsmethoden | 10.3 Anteil an der Gesamtnote |
|--|---|--------------------------|-------------------------------|
| 10.4 Vorlesung   | Korrektter Umgang mit den Grundbegriffen der objektorientierten Programmierung, Verständnis der Funktionsweise von C++ Sprachkonstrukten. | Schriftliche Prüfung     | 30%                           |
| 10.5 Seminar   | Fähigkeit Objekt Orientierten C++ Programmcode zu schreiben.  | praktische Prüfung       | 30%                           |
| 10.6 Labor   | Korrektheit der abgegebenen C++ Programme und die Fähigkeit sie zu erklären.  | Labor Arbeiten           | 40%                           |
| 10.7 Minimale Leistungsstandards   |   |                          |                               |
| <ul style="list-style-type: none"> <li>• Jede Laboraufgabe muss mit mindestens 5 bewertet sein.</li> <li>• Für das Bestehen der praktischen Prüfung muss die Mindestnote 5 erreicht werden.</li> <li>• Für das Bestehen der schriftlichen Prüfung muss die Mindestnote 5 erreicht werden.</li> <li>• Nur die Endnote wird auf ganze Zahl gerundet.</li> <li>• Erforderliche Anwesenheit beim Seminar: 75%. Erforderliche Anwesenheit beim Labor: 90%.</li> </ul> |   |                          |                               |

Ausgefüllt am:

Vorlesungsverantwortlicher

Seminarverantwortlicher

3. Juli 2023

Dominik Knoll, PhD

Dominik Knoll, PhD

Genehmigt im Department am:

Departmentdirektor

.....